IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	RICHARD K ERRICKSON, ET AL.	)
		) Group Art Unit: 2187
Serial No.:	10/723,044	)
		)
Filed:	November 25, 2003	) Examiner: Nguyen, T.
		)
For:	MEMORY MAPPED INPUT/OUTPUT OPERATIONS	)
		) Confirmation No.: 7804
		)

AMENDMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

In response to the Office Action mailed March 13, 2006, please amend the application as follows:

AMENDMENTS TO THE CLAIMS

1. (currently amended) A method of performing memory mapped input output operations to an alternate address space comprising:

establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store data in accordance with ~~a definition of a z/Architecture~~ resource address designation, said resource address designation configured for decomposition thereof such that said first memory mapped input output alternate address space associated with said adapter is accessible;

establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data in accordance with said ~~definition(s) of said z/Architecture~~resource address designation; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space.

2. (original) The method of Claim 1 wherein said first alternate address space is not a partition of a main address space from which said issuing process is executing.

3. (currently amended) The method of Claim 1 wherein said process issuing said at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space operates in a problem state of a machine.

4. (currently amended) The method of Claim 1 wherein said execution includes said at least one of said store and load with an allocated resources associated first alternate address space.

5. (currently amended) The method of Claim 1 wherein said problem state

corresponds to a least privileged execution state. ~~in said z/Architecture~~

6. (currently amended) The method of Claim 1 wherein said first alternate address space is associated with an adapter ~~and alleviates use of a main address space of said process or of another adapter~~.

7. (original) The method of Claim 1 wherein at least one of said first instruction and said second instruction is executed without supervisory state intervention.

8. (cancelled)

9. (currently amended) The method of Claim 1 further including a second alternate address space associated with a second adapter.

10. (original) The method of Claim 9 wherein a storage location in said first alternate address space maps to a different address than the same location in said second alternate address space.

11. (original) The method of Claim 1 wherein said adapter includes address spaces as partitions of said alternate address space.

12. (original) The method of Claim 11 wherein said multiple address spaces are governed by at least one of a resource type and storage area types associated with said adapter.

13. (currently amended) A method of performing memory mapped input output operations to an alternate address space comprising:

establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store in accordance with a

resource address designation, said resource address designation configured for decomposition thereof such that said first memory mapped input output alternate address space associated with said adapter is accessible;

establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data in accordance with said resource address designation; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space and operates in a problem state of a machine.

14. (currently amended) The method of Claim 13 wherein said problem state corresponds to a least privileged execution state. ~~in said z/Architecture~~

15. (cancelled)

16. (original) The method of Claim 13 further including a second alternate address space associated with a second adapter.

17. (currently amended) The method of Claim 16 wherein a storage location in said first alternate address space maps to a different address than the same location in said second alternate address space.

18. (currently amended) A storage medium encoded with a machine-readable computer program code, said code including instructions for causing a computer to implement a method of performing memory mapped input output operations to an alternate address space, the method comprising:

establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store data in accordance with a resource address designation, said resource address designation configured for

decomposition thereof such that said first memory mapped input output alternate address space associated with said adapter is accessible~~definition of a z/Architecture~~;

establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data in accordance with said ~~definition(s) of said z/Architecture~~resource address designation; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space.

19. (currently amended) A system for performing memory mapped input output operations to an alternate address space comprising:

a means for establishing a first instruction directed to a first memory mapped input output alternate address space associated with an adapter to store data in accordance with a resource address designation, said resource address designation configured for decomposition thereof such that said first memory mapped input output alternate address space associated with said adapter is accessible~~definition of a z/Architecture~~;

a means for establishing a second instruction directed to said first memory mapped input output alternate address space associated with an adapter to load data in accordance with said definition(s) of ~~said z/Architecture~~resource address designation; and

wherein a process issues at least one of said first instruction and said second instruction and thereby causes execution of at least one of said store and load with said first alternate address space.

REMARKS

Reconsideration of the instant application is respectfully requested. The present submission is responsive to the Office Action of March 13, 2006, in which claims 1-19 are presently pending. Each of claims 1-19 has been rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter regarded as the invention, as more specifically set forth on page 2 of the Office Action. Claim 6 has been rejected under 35 U.S.C. §112, first paragraph, as failing to comply with the enablement requirement as set forth on page 3 of the Office Action. With regard to the art of record, claims 1-8, 11-15, 18 and 19 have been rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent 6,598,144 to Bailey, et al. In addition, claims 9, 10, 16 and 17 have been rejected under 35 U.S.C. §103(a) as being unpatentable over Bailey. For the following reasons, however, it is respectfully submitted that the present application is now in condition for allowance.

As an initial matter, claims 3-5, 14 and 17 have been corrected so as to provide missing periods, and Applicants respectfully request withdrawal of the objections thereto.

With regard to the outstanding §112, second paragraph rejections, claims 1, 5, 14, 18 and 19 are amended so as to delete the term "z/Architecture." Accordingly, the §112, second paragraph rejections have now been rendered moot, and it is respectfully requested that the same be withdrawn. Furthermore, claims 8 and 15 have been cancelled, rendering the §112, second paragraph rejections thereto moot. With regard to the §112, first paragraph rejection of claim 6, the language "and alleviates use of a main address space of said process or of another adapter" has now been deleted therefrom. Accordingly, this rejection has now also been overcome.

Finally, with regard to each of the outstanding §102 and §103 claim rejections over the art of record (Bailey), independent claims 1, 13 18 and 19 have been amended to

more particularly point out and recite the presently disclosed resource address translation process through a resource address designation (RAD). In particular, resource address translation is a process by which a RAD is decomposed such that the address apace of a memory-mapped adapter can be accessed. Support for this amendment is found at least in paragraphs [0056], [0057], and [0066]-[0070] of the specification, as well as in Figures 2A-2C and Figure 3 of the drawings.

In contrast, Bailey discloses only conventional dynamic virtual-to-physical address translation schemes. See for example, column 4, lines 57-67 of Bailey stating that:

"In particular, the operating system resource 26 assigns a virtual address space 40 for use by the verbs consumer processes 22 the virtual address space 40 is uniquely mapped by an address translator 28 to physical address space 33, including the memory mapped I/O address space 32 and system memory mapped address space 34, for each of the verbs consumer processes 22 based on respective translation map entries 36 stored within a translation table, illustrated in FIG. 4 as translation table 38."

Further, Bailey states in column 5, lines 8-14 that "...[t]he address translator 28 maps the virtual address 42 to a physical address 33 based on retrieving the corresponding unique translation map entry 36 storing a corresponding unique mapping value 46 for the identified consumer process 22, implemented for example as a physical address offset relative to other mapping values based on the prescribed size of the page 40..." and in column 5, lines 33-36 that "...[t]he address translator 28 is implemented as part of the processor core, and is configured for translating addresses 42 from the virtual address space 40 to the physical address space 33, based on identifying a process 22 under execution and based on the corresponding translation map entry 36 assigned to the identified process..."

In other words, the above cited portions of Bailey indicate that the disclosed address translation scheme therein is based wholly on dynamic addressing associated with, for example, Intel-type processor architectures. (see also col. 5, lines 48-51 referring to "x86 based architectures" and "CR3" control registers)

Therefore, since Bailey fails to teach or disclose the claimed resource address designation, each of the outstanding §102 and §103 rejections of the remaining claims have been overcome, and Applicants respectfully request withdrawal of the same.

For the above stated reasons, it is respectfully submitted that the present application is now in condition for allowance. No new matter has been entered and no additional fees are believed to be required. However, if any fees are due with respect to this Amendment, please charge them to Deposit Account No. 06-1130 maintained by Applicants' attorneys.

Respectfully submitted,
RICHARD K ERRICKSON, ET AL.

CANTOR COLBURN LLP
Applicants' Attorneys

By _____
Sean F. Sullivan
Registration No. 38,328
Customer No. 46429

Date: May 24, 2006
Address: 55 Griffin Road South, Bloomfield, CT 06002
Telephone: (860) 286-2929